```
=> d his

     (FILE 'USPAT' ENTERED AT 14:01:18 ON 11 DEC 1997)
L1         42572 S INTERCEPT?
L2         12200 S L1/AB,TI,CLM
L3           902 S L1 (P) NETWORK?
L4           165 S L3/AB,CLM,TI
L5          5741 S SERVER#
L6            46 S L3 (P) L5
L7             8 S L6/AB,CLM,TI
L8           220 S (INTERCEPT? (5A) REQUEST#)
L9            58 S L8 (P) (ROUT? OR REDIRECT?)
L10            6 S L6 (P) L9
L11         1114 S (PLURALITY OR MULTIPLE OR SEVERAL) (5A) SERVER#
L12            1 S L9 (P) L11
L13          153 S (EXTERNAL? (5A) SERVER#)
L14            1 S L9 (P) L13
```

US PAT NO: 5,680,303 :IMAGE AVAILABLE: L10: 1 of 6
TITLE: Communication device sharing on a local area network

DETDESC:

DETD(10)

For . . . the I/O manager 16, 28 processes FO requests which may include requests directed to the X.25 card 36. The NT **redirector** 20 **intercepts request** for non-local (i.e. shared devices). It then "**redirects**" them to the machine on which the device actually resides. The **Network** Transport 22, 34 at each machine is responsible for **routing** and controlling **network** traffic across the local area **network** 24. The request travels across the local area **network** to the target or **server** machine. Next, the request goes to the NT **Server** 32 which is responsible for handling all the messages sent to it by **Redirectors** on all other machines. It determines whether the device has been started, whether the user who sent the request has. . . request which is serviced by the X.25 device driver 28 then travels back the way it came, through the NT **Server**, back to the NT **Redirector**, and eventually to the application that initiated the request 12.

US PAT NO: 5,633,999 :IMAGE AVAILABLE: L10: 2 of 6
TITLE: Workstation-implemented data storage re-routing for server fault-tolerance on computer networks

DETDESC:

DETD(55)

Referring . . . schematically one possible software processing sequence according to the invention, as it may be executed at a workstation on a **server**-redundant **network**, or at a client of multiple **servers**, a data-related, **routed**, drive **request** from an application is **intercepted** via a suitable workstation operating system interrupt 40, for example, DOS interrupt 21. Protected **route** query 42 checks whether the requested drive **route** was specified for monitoring at initialization of the inventive data-access protection software.

US PAT NO: 5,627,829 :IMAGE AVAILABLE: L10: 3 of 6
TITLE: Method for reducing unnecessary traffic over a computer network

DETDESC:

DETD(67)

It is also possible to reduce **network** traffic by synthesizing certain responses such as **routing** information request responses. More specifically, in a typical transmission system, if a first node wishes to establish a connection to a second node, the first node issues a **routing** information request packet to get the address of a **router** to which it should send subsequent data packets. With wireless WAN **networks** there is typically one host computer connected to the **network** to which all of the mobile clients communicate, consequently, the **router** request is unnecessary as all traffic must pass through

this host computer. The optimization layer can **intercept** an incoming **router request** packet and generate an appropriate request response packet as if the response packet had come from a **router**. This operation is illustrated in FIGS. 17B and 17C. FIG. 17B illustrates a **routing** information packet (RIP) transmission between a client node 104 located on a wireless **network** (not shown) and a **server** node 124 located on an enterprise **network** (not shown) when RIP response synthesis is not used. The RIP requests and RIP responses are shown as a function. . .

DETDESC:

DETD(70)

In . . . with one aspect of the invention and, as described above, there is typically one host computer connected to the wireless **network** to which all of the mobile clients communicate, consequently, the optimization layer 1710 can **intercept** an incoming **router request** packet. This latter operation is shown in FIG. 17C which illustrates a RIP request packet transmission which is handled internally within the client node 104. In FIG. 17C, vertical line 1706 represents the standard protocol stack in the **server** node 124 and vertical line 1708 represents the inventive optimization module in the **server** node 124. Vertical lines 1710 and 1712 represent the optimization module and the standard protocol stack in the client node. . .

US PAT NO: 5,537,585 IMAGE AVAILABLE: L10: 4 of 6
TITLE: Data storage management for network interconnected processors

DETDESC:

DETD(54)

As . . . and shadow volumes 65 have their physical storage location identification written into a secondary storage directory 531 in the file **server** 41. The placeholder entry in directory 511 the file **server** 41 points to this secondary storage directory entry. Thus, the processor 21 at step 801 requests access to this migrated data file and this **request** is **intercepted** at step 802 by a trap or interface 711 in the file **server** 41. The trap can utilize hooks in the file system 41 to cause a branch in processing to the storage **server** agent 121 or a call back **routine** can be implemented that allows the storage **server** agent 121 to register with the file system 41 and be called when the data file request is received from the processor 21. In either case, the trapped request is forwarded to storage **server** agent 121 to determine whether the requested data file is migrated to secondary storage 52. This is accomplished by storage **server** agent 121 at step 803 reading directory 511 to determine the location of the requested data file. If a placeholder entry is not found stored in directory 511 at step 805, control is returned to the file **server** 41 at step 806 to enable the file **server** 41 to read the directory entry that is stored in directory 511 for the requested data file. The data stored in this directory entry enables the file **server** 41 to retrieve the requested data file from the data storage device 31 on which the requested data file resides. If at step 805, storage **server** agent 121 determines, via the presence of a placeholder entry, that the requested data file has been migrated to secondary storage 52, storage **server** agent 121 at step 807 creates a data file recall request and transmits this request together with the direct access secondary storage pointer key stored in the placeholder entry via **network** 1 to storage **server** 50. At step 808, operations kernel 501 uses systems services 505 which uses the pointer key to directly retrieve the. . . in the data storage device identified in the secondary storage directory 531 and places the retrieved data file on the **network** 1 for transmission to the file

server 41 and volume 31 that originally contained the requested data file. Systems services 5[redacted] of operations kernel 501 then [redacted]dates. . . the secondary storage directory 531 and the directory 511 to indicate that the data file has been recalled to the **network** volume. At step 811, control is returned to file **server** 41, which reads directory 511 to locate the requested data file. The directory 511 now contains information that indicates the. . . file on data storage device 31. The processor 21 can then directly access the recalled data file via the file **server** 41.

US PAT NO: 5,446,736 :IMAGE AVAILABLE: L10: 5 of 6
TITLE: Method and apparatus for connecting a node to a wireless network using a standard protocol

DETDESC:

DETD(67)

It is also possible to reduce **network** traffic by synthesizing certain responses such as **routing** information request responses. More specifically, in a typical transmission system, if a first node wishes to establish a connection to a second node, the first node issues a **routing** information request packet to get the address of a **router** to which it should send subsequent data packets. With wireless WAN **networks** there is typically one host computer connected to the **network** to which all of the mobile clients communicate, consequently, the **router** request is unnecessary as all traffic must pass through this host computer. The optimization layer can **intercept** an incoming **router request** packet and generate an appropriate request response packet as if the response packet had come from a **router**. This operation is illustrated in FIGS. 17B and 17C. FIG. 17B illustrates a **routing** information packet (RIP) transmission between a client node 104 located on a wireless **network** (not shown) and a **server** node 124 located on an enterprise **network** (not shown) when RIP response synthesis is not used. The RIP requests and RIP responses are shown as a function. . .

DETDESC:

DETD(70)

In . . . with one aspect of the invention and, as described above, there is typically one host computer connected to the wireless **network** to which all of the mobile clients communicate, consequently, the optimization layer 1710 can **intercept** an incoming **router request** packet. This latter operation is shown in FIG. 17C, which illustrates a RIP request packet transmission which is handled internally within the client node 104. In FIG. 17C, vertical line 1706 represents the standard protocol stack in the **server** node 124 and vertical line 1708 represents the inventive optimization module in the **server** node 124. Vertical lines 1710 and 1712 represent the optimization module and the standard protocol stack in the client node. . .

US PAT NO: 5,051,926 :IMAGE AVAILABLE: L10: 6 of 6
TITLE: System wide local copy management of screen copy printing

DETDESC:

DETD(22)

Thus it will be appreciated that the described preferred system allows a local copy function to be performed in a distributed **network** in which terminals can be logically in session with (running applications on) a remote processor. The I/O module acts as a virtual terminal which **intercepts** print **requests** from real terminals and passes these

requests onthe VTAM Local Copy **Server** for processing. The **server** application provides the data formatting and **routing** necessary to implement the local copy function. As printers on the system are used "as needed" they must be leftavailable. . .

=> d 1-6 ti

US PAT NO: 5,680,303 :IMAGE AVAILABLE: L10: 1 of 6
TITLE: Communication device sharing on a local area network

US PAT NO: 5,633,999 :IMAGE AVAILABLE: L10: 2 of 6
TITLE: Workstation-implemented data storage re-routing for server fault-tolerance on computer networks

US PAT NO: 5,627,829 :IMAGE AVAILABLE: L10: 3 of 6
TITLE: Method for reducing unnecessary traffic over a computer network

US PAT NO: 5,537,585 :IMAGE AVAILABLE: L10: 4 of 6
TITLE: Data storage management for network interconnected processors

US PAT NO: 5,446,736 :IMAGE AVAILABLE: L10: 5 of 6
TITLE: Method and apparatus for connecting a node to a wireless network using a standard protocol

US PAT NO: 5,051,926 :IMAGE AVAILABLE: L10: 6 of 6
TITLE: System wide local copy management of screen copy printing

=> d 1-8 ti,kwic

US PAT NO: 5,577,105 :IMAGE AVAILABLE: L7: 1 of 8
TITLE: Telephone call routing and switching techniques for data communications

CLAIMS:

CLMS(4)

4. In a system comprising a remotely located modem, said modem linked to a **network** access **server** over a telephone line, said **network** access **server** linked to a host computer system, a method for processing a credit card transaction comprising the steps of:
initiating a call from said modem to said **network** access **server**;
inputting predetermined control signals onto said telephone line carrying said credit card transaction, wherein said control signals comprise multifrequency tones;
extracting said. . .
said modem, said step of spoofing comprising the steps of
(i) receiving a first signal from said modem at said **network** access **server** and passing said first signal to said host computer system;
(ii) responsively sending a second signal from said **network** access **server** to said modem, causing said modem to initiate transmission of said credit card transaction;
(iii) said host computer sending a third signal to said **network** access **server** in response to said first signal;
(iv) **intercepting** said third signal from said host computer system;
whereby the communication processing time of said credit card transaction is reduced.

US PAT NO: 5,537,585 :IMAGE AVAILABLE: L7: 2 of 8
TITLE: Data storage management for network interconnected processors

CLAIMS:

CLMS(27)

27. The system of claim 26 wherein said storage **server** means comprises:
means, located in each of said file **servers**, for **intercepting** a call at a selected file **server** to data files that have been stored in said file **server**; and
means, responsive to said data written in said **network** directory indicating that said requested data file has been migrated to said secondary storage means, for recalling said requested data file from said secondary storage means to said file **server**.

CLAIMS:

CLMS(68)

68. The method of claim 67 further comprising the steps of:
intercepting a call at a selected file **server** to data files that have been stored in said file **server**; and
recalling, in response to said data written in said **network**

directory indicating that said requested data file has been migrated to said secondary storage means, said requested data file from said secondary storage element to said file **server**.

US PAT NO: 5,491,808 :IMAGE AVAILABLE: L7: 3 of 8
TITLE: Method for tracking memory allocation in network file server

ABSTRACT:
A method for dynamically tracking memory resource allocations/deallocations of a program resident in the memory of a **network** file **server** is disclosed wherein calls to system memory allocation functions are **intercepted** and diverted to memory resident tracker routines, interposed between the caller and the called functions to monitor returns from the. . .

US PAT NO: 5,423,034 :IMAGE AVAILABLE: L7: 4 of 8
TITLE: Network file management with user determined hierarchical file structures and means for intercepting application program open and save commands for inputting and displaying user inputted descriptions of the location and content of files

CLAIMS:

CLMS(8)

8. For use on a computer **network**, the computer **network** including a **server** and computer processor capable of executing an application program, the computer processor coupled to the **server**, the **server** coupled to a plurality of **network** storage devices, each **network** storage device storing a plurality of files in a directory structure, the **server** comprising a **network** access program that sets and checks a set of access privileges for each one of the plurality of files, each one of the directories in the directory structure and each **network** storage device, a method for setting **network** access privileges comprising the steps of:
intercepting control from the application program without exiting the application program when a predetermined command is communicated to the application program;
thereafter, . . . one of the plurality of files, one of the directories in the directory structure or one of the plurality of **network** storage devices;
displaying on an output device coupled to the computer processor the item selected and the set of access privileges for the item;
specifying a new set of access privileges for the item; and
causing the **network** access program of the **server** to alter the set of access privileges for the item.

CLAIMS:

CLMS(29)

29. A **network** access interface controller operateable on a computer processor, the computer processor capable of executing an application program, the computer processor coupled to a **server**, the **server** coupled to a plurality of **network** storage devices, each **network** storage device storing a plurality of files in a directory structure, the **server** comprising a **network** access program that sets and checks a set of access privileges for each one of the plurality of files, each one of the directories in the directory structure and each **network** storage device, the **network** access interface controller comprising:
means for **intercepting** control from the application program without exiting the application program when a predetermined command is received by the application program;

means. . . one of the plurality of files, one of the directories in the directory structure or one of the plurality of **network** storage devices;
means for displaying on an output device coupled to the computer processor the item selected and the set of. . . access privileges for the item;
means for specifying a new set of access privileges for the item; and
means for causing the **network** access program of the **server** to alter the set of access privileges for the item.

CLAIMS:

CLMS(34)

34. A **network** access interface controller operateable on a computer processor, the computer processor capable of executing an application program, the computer processor coupled to a **server**, the **server** coupled to a plurality of **network** storage devices, each **network** storage device storing a plurality of files in a directory structure, the **server** comprising a **network** access program that sets and checks a set of access privileges for each one of the plurality of files, each one of the directories in the directory structure and each **network** storage device, the **network** access interface controller comprising:
means for **intercepting** control from the application program without exiting the application program when a predetermined command is communicated to the application program;
means,. . . one of the plurality of files, one of the directories in the directory structure or one of the plurality of **network** storage devices;
means for displaying on an output device coupled to the computer processor the item selected and the set of. . . access privileges for the item;
means for specifying a new set of access privileges for the item; and
means for causing the **network** access program of the **server** to alter the set of access privileges for the item.

US PAT NO: 5,404,527 :IMAGE AVAILABLE: L7: 5 of 8
TITLE: System and method for remote program load

CLAIMS:

CLMS(5)

5. In a **network** having a workstation with boot processes, a plurality of computer systems capable of providing the functions of a file **server** for the workstation, and an interface processor for communicating with the workstation and the computer systems, the workstation, the computer systems, and the interface processor coupled to the **network**, a method of obtaining boot services from one of the computer systems during the boot process of a workstation on the **network**, wherein during the boot process some ones or all of the computer systems can provide boot services, said method comprising the steps of:
(a) sending, from the workstation, a boot service request on the **network**;
(b) **intercepting**, at the interface processor, said boot service request on the **network**;
(c) selecting, at the interface processor, which ones of the plurality of computer systems can provide boot services,
(d) sending, from the interface processor, the boot service request on the **network** to the selected computer systems, the boot service request being formatted to be received by the selected computer systems;
(e) receiving,. . . the interface processor, a response from a responsive one of said selected computer systems; and

(f) opening a path thro[redacted] the **network** to the responsive computer system for providing b[redacted] services to the workstation.[redacted]

CLAIMS:

CLMS(6)

6. In a **network** having a workstation with a boot process, a plurality of computer systems capable of performing the functions of a file **server**, and an interface processor for communicating with the workstation and the computer systems, the workstation, the computer systems, and the interface processor coupled to the **networks** a method of retrieving informations through the **network**, from a disk storage coupled to a respective one of the plurality of computer systems during the boot process of. . .
computer systems can provide boot services,
(b) sending, from the interface processor, a request for retrieval of the information through the **network**, said request to be received by the selected ones of the plurality of computer systems which are coupled to the **network**;
(c) receiving, at the interface processor, a first response from a responsive one of the selected computer systems;
(d) responsive to the first response, sending, from the interface processor, a confirmation to the responsive computer system on the **network**;
(e) **intercepting**, at the interface processor, a request for the information during the boot process of the workstation from the workstation;
(f) sending,. . .

CLAIMS:

CLMS(8)

8. In a **network** having a workstation with boot processes, a plurality of computer systems capable of performing the functions of a file **server** for the workstation, and an interface processor for communicating with the workstation and the computer systems, the workstation, the computer systems, and the interface processor coupled to the **network**, a method of retrieving information from a disk storage on one of the plurality of computer systems during the boot process of the workstation on the **network**, wherein during the boot process some ones or all Of the computer systems can provide boot services, said method comprising the steps of:
(a) **intercepting**, at the interface processor, a request for information from a disk during the boot process of the workstation;
(b) selecting, at the interface processor, which ones of the plurality of computer systems can provide boot services,
(c) in response to the **intercepted** request, sending, from the interface processor, a request to open a path through the **network** for retrieval of the information, said request to be received by the selected ones of the plurality of computer systems which are coupled to the **network**;
(d) receiving, at the interface processor, a response from a responsive one of the selected computer systems;
(e) sending, from the interface processor, a confirmation to the responsive computer system on the **network**;
(f) sending, from the interface processor, the request for the information to the responsive computer system;
(g) receiving, at the interface processor,. . .

CLAIMS:

CLMS(9)

9. In a **network** having [redacted]kstations with boot processes, a plurality of computer systems capa[redacted]e of providing the functions of [redacted] file **server**, and an interface processor for communicating with the workstation and the computer systems, the workstation, the computer systems, and the interface processor coupled to the **network**, a method of retrieving data from disk storage on one of the plurality of computer systems during the boot process of a workstation on the **network**, wherein during the boot process some ones or all of the computer systems can provide boot services, said method comprising the steps of:

(a) **intercepting**, at the interface processor, a request for program code during the boot process of the workstation;

(b) selecting, at the interface. . . services,

(c) sending, from the interface processor, a request for retrieval of data during the boot process of workstation on the **network**, said request to be received by the selected ones of the plurality of computer systems which are coupled on the **network**;

(d) receiving, at the interface processor, a response from a responsive one of the selected plurality of computer systems;

(e) responsive to the received first response, sending, from the interface processor, a confirmation to the responsive computer system on the **network**;

(f) sending, from the interface processor, the request for the program code to the responsive computer system on the **network**;

(g) receiving, at the interface processor, a second response from the responsive computer system through the **network**, the second response containing the requested computer code;

(h) retrieving, at the interface processor, the program code from the second response;. . .

US PAT NO: 5,367,698 :IMAGE AVAILABLE: L7: 6 of 8
TITLE: Network file migration system

CLAIMS:

CLMS(13)

13. A **networked** digital data processing system according to claim 12, wherein the migration file **server** comprises **intercept** means, coupled to said selected client device, for **intercepting** and handling selected file access requests transmitted between elements of the client filesystem thereof, the **intercept** means including

A. detection means for detecting selected file access requests transmitted between elements of such client filesystem,

B. means, coupled. . .

CLAIMS:

CLMS(14)

14. A **networked** digital data processing system according to claim 13, wherein

A. the migration file **server** includes means for storing a set of digital flags representative of the state of data files stored in the migration storage means, and

B. the control means of the **intercept** means includes means updating selected ones of said digital flags upon transfer of the data portion of a requested data file in response to an **intercepted** access request.

US PAT NO: 5,305,456 :IMAGE AVAILABLE: L7: 7 of 8
TITLE: Apparatus and method for computer system integrated security

ABSTRACT:
Computer system security apparatus and method extends control of operating system security into an application. An application control

component **intercepts** and interprets the security processing of application security software of an application. An interface member formats calls to the operating. . . platforms/computers. the apparatus and method extends control of the operating system of a mainframe into those applications running on devices **networked** to the mainframe. The mainframe thus functions as a security **server** which distributes security functionality to numerous applications running on diverse platforms.

US PAT NO: 4,866,706 : IMAGE AVAILABLE: L7: 8 of 8
TITLE: Token-passing local area network with improved throughput

ABSTRACT:
A controller is disclosed for use in a local area **network** preferably of the type having a token-passing protocol. The controller modifies the contents of a data packet or message as it moves on the **network** media, thereby to **intercept**, modify and redirect the data message to nodes other than the originally addressed node. In accordance with one embodiment of the invention, the controller is associated with a high-information node (e.g., a file **server**) connected in a **network** with a plurality of low-information nodes. The controller modifies the token address transmitted with the token packet from the h

=> d 1-8 ti

US PAT NO: 5,577,105 :IMAGE AVAILABLE: L7: 1 of 8
TITLE: Telephone call routing and switching techniques for data communications

US PAT NO: 5,537,585 :IMAGE AVAILABLE: L7: 2 of 8
TITLE: Data storage management for network interconnected processors

US PAT NO: 5,491,808 :IMAGE AVAILABLE: L7: 3 of 8
TITLE: Method for tracking memory allocation in network file server

US PAT NO: 5,423,034 :IMAGE AVAILABLE: L7: 4 of 8
TITLE: Network file management with user determined hierarchical file structures and means for intercepting application program open and save commands for inputting and displaying user inputted descriptions of the location and content of files

US PAT NO: 5,404,527 :IMAGE AVAILABLE: L7: 5 of 8
TITLE: System and method for remote program load

US PAT NO: 5,367,698 :IMAGE AVAILABLE: L7: 6 of 8
TITLE: Network file migration system

US PAT NO: 5,305,456 :IMAGE AVAILABLE: L7: 7 of 8
TITLE: Apparatus and method for computer system integrated security

US PAT NO: 4,866,706 :IMAGE AVAILABLE: L7: 8 of 8
TITLE: Token-passing local area network with improved throughput